

Ohjeet: Merkitse selvästi kunkin osaamisalueen osalta, mikä annetuista vaihtoehdoista kuvaa parhaiten joukkueen suoritusta. Mikäli suoritus jää näiden alle, merkitään EI (Ei suoritusta). Kirjaa kommenttikenttiin mahdollisimman monta kehitysehdotusta, sekä arvioinnissa joukkueen eduksi huomioitavaa havaintoa. Ympyröi lopuksi alareunan vaihtoehdoista joukkueen vahvin osaamisalue. Tämä dokumentti on perustuu englanninkieliseen alkuperäisjulkaisuun: Ristiriitatapauksissa pätee englanninkielinen teksti.

	Aloitteleva	Kehittyvä	Taitava	Esimerkillinen	
Mekaniikan suunnittelu	Kestävyys Rakenteen lujuus; kyky kestää kilpailun rasituksia, mekaniikka				
	EI	varsin herkkä särkymään; voi rikkoutua pahoin	jotain pientä pitää korjata säännöllisesti	vikoja / korjattavaa esiintyy vain harvoin	vaikuttava rakenne; ei tarvitse korjailia
	Rakenteen tehokkuus Taloudellinen osien ja ajan käyttö; helppo korjata ja muokata				
	EI	osia on käytetty paljon yli tarpeen, tai rakenne on hidas muokata / korjata	osia on käytetty yli tarpeen, tai rakenne on hidas muokata / korjata	asianmukainen rakenne, muokattavissa ja korjattavissa	tehokkaasti rakennettu robotti, joka on muokattavissa ja korjattavissa nopeasti
Comments	Rakenteen toimivuus Robotin liikkeet ja mekanismit toimivat suoritettaviin tehtäviin nähden sopivalla nopeudella, voimalla ja tarkkuudella (käyttövoima ja toteutus)				
	EI	nopeus, voima ja tarkkuus ovat useimpiin tehtäviin nähden epätasapainossa	nopeus, voima ja tarkkuus ovat muutamaan tehtävään nähden epätasapainossa	nopeus, voima ja tarkkuus ovat useimpiin tehtäviin nähden tarkoituksenmukaisia	nopeus, voima ja tarkkuus ovat kaikkiin tehtäviin nähden tarkoituksenmukaisia
Ohjelmointi	Ohjelmoinnin laatu Ohjelmat ovat tarkoituksenmukaisia ja niillä saavutetaan toistettavia tuloksia, olettaen ettei robottiin tule mekaanisia vikoja				
	EI	tavoitteet saattavat jäädä saavuttamatta JA robotti saattaa toimia	tavoitteet saattavat jäädä saavuttamatta TAI robotti saattaa toimia	saavuttanee tavoitteet toistuvasti	saavuttaa tavoitteet joka kerta
	Ohjelmoinnin tehokkuus Ohjelmat ovat modulaarisia, selkeitä ja ymmärrettäviä				
	EI	tarpeettoman pitkästi kirjoitettua koodia, josta on vaikea saada selkoa	tarpeettoman pitkästi kirjoitettua koodia, josta on hankalaa saada selkoa	tarkoituksenmukainen koodi, jota on helppo ymmärtää	suoraviivainen tarkoituksenmukainen koodi, jota kaikki ymmärtävät helposti
Comments	Automaattiset toiminnot/Navigation Robotin kyvykkyys havainnoida ja hyödyntää ympäristöään, sekä toimia ja liikkua itsenäisesti tältä pohjalta (riippumattomuus käyttäjän toimenpiteistä ja / tai ohjelman ajoituksista)				
	EI	suoritukset perustuvat säännöllisesti käyttäjän tekemään tähtäykseen JA robotin palautamiseen	suoritukset perustuvat säännöllisesti käyttäjän tekemään tähtäykseen TAI robotin palautamiseen	Robotti liikkuu ja toimii toistuvasti aiotulla tavalla, vain satunnaista käyttäjän toimenpiteitä	Robotti liikkuu ja toimii joka kerta aiotulla tavalla, ilman käyttäjän toimenpiteitä
Strategia & Ideointi	Suunnitteluprosessi Joukkue osaa kuvailla robotin suunnittelussa käyttämänsä ideointi, kehitys ja testaus prosessin pääkohdat (koskee ohjelmointia sekä mekaniikan suunnittelua)				
	EI	järjestelmällisyyteen JA perusteluihin on syytä kiinnittää enemmän huomiota	järjestelmällisyyteen TAI perusteluihin on syytä kiinnittää enemmän huomiota	järjestelmällinen, hyvin perusteltu	järjestelmällinen, hyvin perusteltu ja dokumentoitu
	Pelistrategia Joukkueen pelistrategia on harkiten määritelty ja selkeä				
	EI	joukkueella ei ole selkeitä tavoitteita, EIKÄ selkeää strategiaa tehtävien suorittamiseen	joukkueella ei ole selkeitä tavoitteita TAI selkeää strategiaa tehtävien suorittamiseen	joukkue on määritelty tavoitteensa ja sillä on selkeä strategia niiden saavuttamiseksi	joukkueella on selkeä strategia useimpien / kaikkien tehtävien suorittamiseksi
Comments	Kekseliäisyys Uusi, ainutlaatuinen, tai odottamaton ominaisuus, idea (esim. mekaniikkaa, ohjelmia, strategioita tai sovelluksia), jotka ovat hyödyllisiä tiettyjen tehtävien suorittamisessa				
	EI	omaperäinen ratkaisu (ratkaisuja) joka ei tuo lisäarvoa tai mahdollisuuksia	omaperäinen ratkaisu (ratkaisuja) joka tuo jotakin lisäarvoa tai mahdollisuuksia	omaperäinen ratkaisu (ratkaisuja) joka lisää mahdollisuuksia merkittävästi	omaperäinen ratkaisu (ratkaisuja) joka tuo merkittävästi lisäarvoa
	Vahvin osa-alue:	Mekaniikan suunnittelu	Ohjelmointi	Strategia & Ideointi	

Directions: For each skill area, clearly mark the box that best describes the team's accomplishments. If the team does not demonstrate skill in a particular area, then put an 'X' in the first box for Not Demonstrated (ND). Please provide as many written comments as you can to acknowledge each team's hard work and to help teams improve. When you have completed the evaluation, please circle the team's areas of strength.

		Beginning	Developing	Accomplished	Exemplary	
Mechanical Design	Durability	Evidence of structural integrity; ability to withstand rigors of competition				
	N D	quite fragile; breaks a lot	frequent or significant faults/repairs	rare faults/repairs	sound construction; no repairs	
	Mechanical Efficiency	Economic use of parts and time; easy to repair and modify				
	N D	excessive parts or time to repair/modify	inefficient parts or time to repair/modify	appropriate use of parts and time to repair/modify	streamlined use of parts and time to repair/modify	
Programming	Mechanization	Ability of robot mechanisms to move or act with appropriate speed, strength and accuracy for intended tasks (propulsion and execution)				
	N D	imbalance of speed, strength and accuracy on most tasks	imbalance of speed, strength and accuracy on some tasks	appropriate balance of speed, strength and accuracy on most tasks	appropriate balance of speed, strength and accuracy on every task	
	Programming Quality	Programs are appropriate for the intended purpose and would achieve consistent results, assuming no mechanical faults				
	N D	would not achieve purpose AND would be inconsistent	would not achieve purpose OR would be inconsistent	should achieve purpose repeatedly	should achieve purpose every time	
Strategy & Innovation	Programming Efficiency	Programs are modular, streamlined, and understandable				
	N D	excessive code and difficult to understand	inefficient code and challenge to understand	appropriate code and easy to understand	streamlined code and easy for anyone to understand	
	Automation/Navigation	Ability of the robot to move or act as intended using mechanical and/or sensor feedback (with minimal reliance on driver intervention and/or program timing)				
	N D	frequent driver intervention to aim AND retrieve robot	frequent driver intervention to aim OR retrieve robot	robot moves/acts as intended repeatedly w/ occasional driver intervention	robot moves/acts as intended every time with no driver intervention	
Comments	Design Process	Ability to develop and explain improvement cycles where alternatives are considered and narrowed, selections tested, designs improved (applies to programming as well as mechanical design)				
	N D	organization AND explanation need improvement	organization OR explanation need improvement	systematic and well- explained	systematic, well-explained and well-documented	
	Mission Strategy	Ability to clearly define and describe the team's game strategy				
	N D	no clear goals AND no clear strategy	no clear goals OR no clear strategy	clear strategy to accomplish the team's well defined goals	clear strategy to accomplish most/all game missions	
Comments	Innovation	Creation of new, unique, or unexpected feature(s) (e.g. designs, programs, strategies or applications) that are beneficial in performing the specified tasks				
	N D	original feature(s) with no added value or potential	original feature(s) with some added value or potential	original feature(s) with the potential to add significant	original feature(s) that add significant value	
	Strengths:		Mechanical Design	Programming	Strategy & Innovation	